

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平4-242829

(43) 公開日 平成4年(1992)8月31日

(51) Int.Cl.⁵

G 0 6 F 9/06

識別記号

4 4 0 C 7927-5B

庁内整理番号

F I

技術表示箇所

審査請求 未請求 請求項の数 1 (全 8 頁)

(21) 出願番号 特願平3-457

(22) 出願日 平成3年(1991)1月8日

(71) 出願人 000005223

富士通株式会社

神奈川県川崎市中原区上小田中1015番地

(72) 発明者 奥住 信明

神奈川県川崎市中原区上小田中1015番地

富士通株式会社内

(72) 発明者 村岡 文子

神奈川県川崎市中原区上小田中1015番地

富士通株式会社内

(72) 発明者 吉田 禎彦

神奈川県川崎市中原区上小田中1015番地

富士通株式会社内

(74) 代理人 弁理士 井桁 貞一

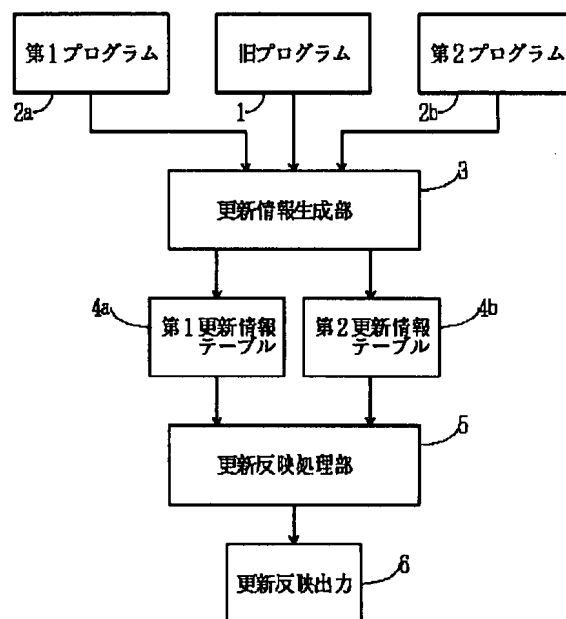
(54) 【発明の名称】 プログラム更新処理装置

(57) 【要約】

【目的】 計算機の原始プログラムの更新処理に関し、シーケンス番号に依存しないで2プログラムの相違を認識でき、それに基づいて2つの独立した更新の一方の更新を他方に反映するプログラム更新処理装置を目的とする。

【構成】 旧プログラムと、該旧プログラムを更新した新プログラムを比較して更新情報を生成するに際し、該旧プログラムと新プログラムとを入力して、該旧プログラムと、該新プログラムとについて、各命令文の文字列の所定の大小順に従って該命令文を順序付けた旧順位テーブルと、新順位テーブルとを生成し、該旧順位テーブルと、該新順位テーブルとに示す順位に従って該旧プログラムの該命令文と、該新プログラムの該命令文とを比較し、両該命令文の該大小関係の所定条件に従って、該旧プログラムに対して削除、挿入及び無変更の該命令文を判別し、該判別結果を参照して、更新情報の生成等を行うように構成する。

本発明の構成を示すブロック図



【特許請求の範囲】

【請求項1】 1以上の命令文の並びによって構成された所与の原始プログラムである旧プログラム(1)と、該旧プログラムに第1の更新を行った第1プログラム(2a)と、第2の更新を行った第2プログラム(2b)とがある場合に、第1の更新を第2プログラムに反映させる処理において、更新情報生成部(3)と、更新反映処理部(5)とを設け、該更新情報生成部(3)は、該旧プログラム(1)、第1プログラム(2a)、及び第2プログラム(2b)を入力して、該旧プログラム(1)と第1プログラム(2a)とについて、第1プログラムを新プログラムとする更新情報生成処理によって第1更新情報テーブル(4a)を生成し、該旧プログラム(1)と第2プログラム(2b)とについて、第2プログラムを該新プログラムとする該更新情報生成処理によって第2更新情報テーブル(4b)を生成し、該更新情報生成処理は、該旧プログラムと該新プログラムとについて、各該命令文の文字列の所定の大小関係に従って該命令文を順序付けた旧順位テーブルと、新順位テーブルとを生成する処理と、該旧順位テーブルと、該新順位テーブルとに示す順位に従って該旧プログラムの該命令文と、該新プログラムの該命令文とを比較し、両該命令文の該大小関係の所定条件に従って、各該命令文について所定の削除、挿入及び無変更を判別する処理と、該判別結果を参照して、該旧プログラムと該新プログラムとの該命令文をプログラム上の配列順に比較することにより、該旧プログラムと該新プログラムの各該命令文の挿入、削除、移動及び無変更を決定して、該決定した状態を表す所定の更新情報テーブルを生成する処理からなり、該更新反映処理部(5)は、第1更新情報テーブル(4a)の削除、挿入及び移動項目と、第2更新情報テーブル(4b)とを照合して、第2プログラム(2b)に第1の更新を反映する所定の制御情報を生成し、所要の第1及び第2更新情報テーブルの情報と共に出力するように構成されていることを特徴とするプログラム更新処理装置。

【発明の詳細な説明】

【0001】

【産業上の利用分野】本発明は、計算機の原始プログラムについて異なる2種の修正が行われた場合に、更新状況を把握し、一方の更新内容を他方の更新されたプログラムに反映する処理を行うための、プログラム更新処理装置に関する。

【0002】

【従来の技術】例えば計算機システムのいわゆるパッケージプログラムについて、原始プログラムのレベルのいわゆるカスタマイズによって、利用者システムごとの修正を行って使用している場合がある。

【0003】このパッケージプログラムが機能アップ等で改版されたような場合には、カスタマイズしたプログラムについても機能アップを反映することが要求される。この例のように、ある原始プログラムに対して独立

に異なる修正が行われた場合に、両修正を共に矛盾無く適用した新プログラムを生成する必要がある場合があり、そのために一方の修正を他方にも反映するように、他方の新プログラムを更に修正する方法がとられる。

【0004】そのような修正は、それぞれ元のプログラムとの対応を維持して、修正結果をチェックし易くするために、元のプログラムの命令文に付けられたシーケンス番号を保存して、そのプログラムに必要な命令文の削除及び挿入を行う形で行われる。

【0005】従って、前記のような変更を自動処理する場合には、両原始プログラムに付けられているシーケンス番号に基づいて、対応する命令文を比較することにより、挿入、削除、更新等を判別して処理する。

【0006】

【発明が解決しようとする課題】以上のように、両原始プログラムは元のプログラムに対してシーケンス番号を保存するようにして更新しなければならないので、実際に可能な修正が制約される場合が生じる。

【0007】又、前記のような判別方法では命令文の移動を判別できないので、チェックのための資料として十分でなく、人手によるチェックの工数を増大するという問題がある。

【0008】本発明は、シーケンス番号に依存しないで2プログラムの相違を移動を含めて認識でき、それに基づいて2つの独立した更新の一方の更新を他方に反映することができるプログラム更新処理装置を目的とする。

【0009】

【課題を解決するための手段】図1は、本発明の構成を示すブロック図である。図はプログラム更新処理装置の構成であって、1以上の命令文の並びによって構成された所与の原始プログラムである旧プログラム1と、旧プログラム1に第1の更新を行った第1プログラム2aと、第2の更新を行った第2プログラム2bとがある場合に、第1の更新を第2プログラム2bに反映させる処理において、更新情報生成部3と、更新反映処理部5とを設ける。

【0010】更新情報生成部3は、旧プログラム1、第1プログラム2a、及び第2プログラム2bを入力して、旧プログラム1と第1プログラム2aとについて、第1プログラム2aを新プログラムとする更新情報生成処理によって第1更新情報テーブル4aを生成し、該旧プログラムと第2プログラム2bとについて、第2プログラム2bを該新プログラムとする該更新情報生成処理によって第2更新情報テーブル4bを生成する。

【0011】該更新情報生成処理は、該旧プログラム1と該新プログラムとについて、各該命令文の文字列の所定の大小関係に従って該命令文を順序付けた旧順位テーブルと、新順位テーブルとを生成する処理と、該旧順位テーブルと、該新順位テーブルとに示す順位に従って該旧プログラムの該命令文と、該新プログラムの該命令文

とを比較し、両該命令文の該大小関係の所定条件に従って、各該命令文について所定の削除、挿入及び無変更を判別する処理と、該判別結果を参照して、該旧プログラムと該新プログラムとの該命令文をプログラム上の配列順に比較することにより、該旧プログラムと該新プログラムの各該命令文の挿入、削除、移動及び無変更を決定して、該決定した状態を表す所定の更新情報テーブルを生成する処理からなる。

【0012】更新反映処理部5は、第1更新情報テーブル4aの削除、挿入及び移動項目と、第2更新情報テーブル4bとを照合して、第2プログラム2bに第1の更新を反映する所定の制御情報を生成し、所要の第1及び第2更新情報テーブル2a、2bの情報と共に出力する。

【0013】

【作用】本発明の処理装置により、旧プログラムから新プログラムへの修正状況を検出する場合に、両プログラムを命令文の文字列でソートした結果について、ソート順に比較することによって削除、挿入を検出する。

【0014】以上によって無変更と判別された命令文は、完全に無変更で残っているか、単に平行移動したのみの無変更か、他との相対位置の変わる移動かの何れかであるので、前の処理で無変更となった命令文について、プログラム配列の順にそれらを比較することにより、移動を抽出することができる。

【0015】以上のようにして第1及び第2プログラムについて作成する更新情報テーブルに基づいて、第1プログラムに行った更新を第2プログラムにも反映するための制御情報を自動生成する。

【0016】利用者はこの制御情報を、前記の更新情報等を参照してチェックすることにより、必要な補正等を行った上で、この制御情報によって第2プログラムを再更新して必要なプログラムの更新を完了することができる。

【0017】

【実施例】図2は、図1に示す本発明の更新情報生成部3及び更新反映処理部5の処理の内容を、生成する各種テーブルの関係によって示すものである。

【0018】更新情報生成部3は、先ず旧プログラム1、第1プログラム2a、第2プログラム2bから、それぞれ旧ソーステーブル11、第1ソーステーブル12a、第2ソーステーブル12bを作成する。各ソーステーブルは、各原始プログラムの命令文を、プログラム上の配列順に並べて、配列順番号を付けたテーブルであって、例えば図6(a)のような構成とする。

【0019】次に、各ソーステーブルの命令文を、その文字列によってソートし、例えば小さい順に順序付けるものとして、そのソート順にソーステーブルの配列順番号を並べた図6(b)のような順位テーブルを、旧順位テーブル13、第1順位テーブル14a、第2順位テーブル14bのように作成する。

【0020】次に、旧順位テーブル13と第1順位テーブル14aを参照して、2プログラムの該当する命令文を比較する。この最初のマッチング処理によって、両プログラムの同一命令文の対応を示す第1対応テーブル15aを作成すると共に、第1プログラム2aへの修正で、旧プログラム1から削除された命令文等を、旧ソーステーブル11上でマークし、又第1プログラム2aへ挿入された命令文等を第1ソーステーブル12a上でマークする。

【0021】又、旧順位テーブル13と第2順位テーブル14bとを参照する、前記と同様の最初のマッチング処理によって、第2対応テーブル15bを作成しながら、第2プログラム2bへの修正による削除及び挿入命令文を旧ソーステーブル11と、第2ソーステーブル12b上でマークする。

【0022】各対応テーブルは、新旧プログラム間で同じ文字列の命令文の1対1の対応関係を、配列順番号で表すようにしたテーブルであって、例えば図6(c)のように構成する。

【0023】図3は以上に述べた最初のマッチング処理により、各対応テーブル作成等を行う処理の流れの一例を示す図であり、1、jを項番号として、旧順位テーブル13の第1項の配列順番号で指示する旧ソーステーブル11の命令文の文字列と、新順位テーブル(第1順位テーブル14a又は第2順位テーブル14b)の第j項の配列順番号で指示する新ソーステーブル(第1ソーステーブル12a又は第2ソーステーブル12b)の命令文の文字列とを比較して処理するものとする。

【0024】先ず図3の処理ステップ20でiとjを1に初期設定した後、処理ステップ21でi、jで定まる新旧プログラムの命令文(以下において新命令文及び旧命令文という)の文字列を比較し、等しければ処理ステップ22で対応テーブル(第1対応テーブル15a又は第2対応テーブル15b)に新旧命令文の配列順番号の対応を記録し、処理ステップ23でiとjを共に+1して進める。

【0025】旧命令文の方が小さい場合には、旧命令文が削除されたと判定し、処理ステップ24で旧ソーステーブル11の該当命令文に、削除を示すDフラグを記録し、処理ステップ25で対応テーブルの当該旧側の配列順番号に対応する番号を0とし、処理ステップ26でiのみを+1する。

【0026】旧命令文の方が大きい場合には、新命令文が挿入されたと判定し、処理ステップ27で新ソーステーブルの該当命令文に、挿入を示すIフラグを記録し、処理ステップ28で対応テーブルの当該新命令文側の配列順番号に対応する番号に0を記録し、処理ステップ29でjのみを+1する。

【0027】以上の後、処理ステップ30で何れか一方の順位テーブルが終了したか識別し、終了していなければ処理ステップ21に戻って以上の処理を繰り返す。終了の場合には処理ステップ31で、旧順位テーブル13に項が余

っていれば、それらの各項の指示する旧命令にDフラグを記録し、新順位テーブルが余っていれば、該当する新命令にIフラグを記録する。

【0028】又、各順位テーブル13、14a、14bについて、tを順位テーブルの項番号とし、図4に示すように、処理ステップ32でtを当該順位テーブルの最終項の値に初期設定して、処理ステップ33で順位テーブルの第t項の配列順番号で指示される命令文と、第t-1項の配列順番号で指示される命令文とを比較する。

【0029】両命令文が等しければ、処理ステップ34で第t項の配列順番号で指示される命令文にEフラグを記録し、等しくなければ何もせず、処理ステップ36でtを-1して処理ステップ33に戻り、処理ステップ35で識別してtが1になるまで以上を繰り返す。

【0030】以上により、旧ソーステーブル11、第1ソーステーブル12a、第2ソーステーブル12bの所要の命令文にD、I、Eフラグが記録される。D又はIフラグの付いていない命令文は、新旧プログラムで対応のある命令文である。

【0031】ここで更新情報生成処理部3は、新旧ソーステーブルの各命令文を比較し、必要に応じて対応テーブルを参照する2回目のマッチング処理によって、更新情報テーブル（第1更新情報テーブル4a及び第2更新情報テーブル4b）を作成する。更新情報テーブルは、例えば図6(d)に示すように項（処理順に対応）の欄と、フラグ欄と、番号X欄と、プログラム区分欄と、番号Y欄とからなる。

【0032】プログラム区分欄はN及びOによって、番号X欄の配列順番号が新ソーステーブル（N）上の番号か旧ソーステーブル（O）上の番号かを示し、番号Yの値は後述する。

【0033】図5は2回目のマッチング処理の流れの一例を示す図であり、mを新ソーステーブルの配列順番号、nを旧ソーステーブルの配列順番号として、配列順番号nで指示する旧ソーステーブル11の命令文の文字列と、配列順番号mで指示する新ソーステーブルの命令文の文字列とを配列順番号の順に比較して処理する流れを示す。

【0034】先ず図5の処理ステップ40でmとnを1に初期設定した後、処理ステップ41で処理対象の新旧ソーステーブルの配列順番号mか配列順番号nに、前記の処理で設定したフラグがあるか識別し、フラグがある場合には、処理ステップ42でフラグの種類を識別する。

【0035】フラグがD、Iか、又はT（後述）であれば、処理ステップ43でテーブルの次の1項のフラグにこのフラグを設定し、処理ステップ44でプログラム区分を、フラグがDならO、フラグがI及びTならNにし、処理ステップ45で、プログラム区分に対応する配列順番号（即ちそのときのm又はn）を番号Xに設定し、処理ステップ46で番号Yを設定する。

【0036】番号Yには、対応テーブル上で番号Xに対応する番号（例えば番号Xに新プログラムの配列順番号を設定したときは、番号Yは対応テーブルで対応している旧プログラムの配列順番号）を設定する。

【0037】但し、フラグがI及びTの場合にはこの更新情報テーブルの前項の番号Yと同じ値にする。又、フラグがDの場合には、第1プログラムの処理のとき、即ち第1更新情報テーブルの作成時は、そのテーブルの前項の番号Yと同じ値にし、第2プログラムの処理のときは、その項の番号Xと同じ値にする、以上は後述の反映テーブル作成時の処理の便のためである。

【0038】以上で1項の設定を完了し、処理ステップ53でソーステーブルの末尾まで処理したか識別し、未処理があれば処理ステップ54で更新情報テーブルに登録する対象となった方の配列順番号（m又はn又は両方）を+1し、処理ステップ41に戻る。

【0039】処理ステップ41でフラグが無かった場合には、処理ステップ47で新プログラムの配列順番号mの命令文と、旧プログラムの配列順番号nの命令文とを各ソーステーブルから取り出して文字列を比較する。

【0040】その結果等しくなかった場合は、旧プログラムの当該命令文は移動されたものとみなし、処理ステップ48でフラグにFを設定して移動元を示し、処理ステップ49で対応テーブルによりその旧プログラム命令文に対応する新プログラムの命令文を決定して、新ソーステーブルの当該命令文のフラグをTに設定して移動先を示す。

【0041】以上の後、処理ステップ43以降の処理で前記と同様に更新情報テーブルの1項を完成する。但し、フラグFはフラグDと同様に処理するものとする。処理ステップ47で新旧命令文を比較して文字列が等しい場合には、処理ステップ50で更新情報テーブルの次の項で、プログラム区分をNに設定した後、処理ステップ45に進み、以下前記と同様に1項を完成する。従ってこの場合にはフラグ欄は空白となり、無変更の命令文であることを示す。

【0042】前記の処理ステップ42でフラグがEであった場合には、処理ステップ51で新旧命令文を比較し、文字列が等しければ処理ステップ50に進んで前記と同様に処理し、等しくない場合は処理ステップ52で、旧プログラムならフラグをDに変え、新プログラムならIに変えて処理ステップ43に進み、以後前記のように処理を進める。

【0043】以上の処理により、図6(a)の旧プログラムと新プログラムの各ソーステーブルについて、(c)の対応テーブルが作成され、それらから(d)の更新情報テーブルを得る。

【0044】更新情報生成部3は、以上の処理を第1ソーステーブル12aと旧ソーステーブル11について実行して第1更新情報テーブル4aを生成する。又第2ソーステ

ーブル12bと旧ソーステーブル11について実行して第2更新情報テーブル4bを生成する。

【0045】そこで、更新反映処理部5が、第1更新情報テーブル4aからフラグの付いた項を抽出し、フラグのFはD、TはIに変更して差分情報テーブル16を作成し、差分情報テーブル16と第2更新情報テーブル4bとから、第1プログラムの更新内容を第2プログラムに反映するためのデータとなる反映テーブル17を作成する。反映テーブルは例えば図8(d)の構成とする。

【0046】図7は反映テーブル17を作成する処理の流れの一例を示す図であり、1、jを項番号として、第2更新情報テーブルの第1項と、差分情報テーブルの第j項との番号Yの比較によって処理を進めることにより、第2更新情報テーブル又は差分情報テーブルの該当項の内容により反映テーブル17を作成していく処理の流れを示す。

【0047】図7の処理ステップ60で1及びjの値を1に初期設定し、処理ステップ61で第2更新情報テーブル4bの第1項の番号Yと、差分情報テーブルの第j項の番号Yとを比較する。

【0048】その結果、第2更新情報テーブル4bの第1項の番号Yの方が大きくない(小さいか、等しい)場合には、処理ステップ62で第2更新情報テーブル4bの第1項の内容を反映テーブルの1項に設定する。次いで、処理ステップ63で1を+1し、テーブルが終わっていないければ処理ステップ64から処理ステップ61に戻り、新しい1と元のjによって比較をする。

【0049】処理ステップ61の比較で第2更新情報テーブル4bの第1項の番号Yの方が大きければ、処理ステップ65で差分情報テーブルの第j項を、フラグDをC、IをRに変更して反映テーブルの次の項に設定する。

【0050】その項について、処理ステップ66でプログラム区分欄をHにして処置を要するデータの項であることを示し、処理ステップ67で1つ前の項のプログラム区分をXにしてH項の前であることを示す。

【0051】処理ステップ68で番号Xと番号Yに1つ前の項の値を引き継いでその項を完成した後、処理ステップ69でjを+1して処理ステップ64を経て処理ステップ61に戻り、以上の処理を繰り返す。

【0052】図8は以上の処理の各結果の例を示し、図8(a)に例示するソーステーブルに基づいて、(b)に示す各更新情報テーブルが作成され、その第1更新情報テーブルから(c)に示す差分情報テーブルが抽出される。(b)の第2更新情報テーブルと(c)の差分情報テーブルから、前記の処理により(d)に示す反映テーブルを得る。この反映テーブルを参照することにより、更新反映出力6の1つとして、例えば図8(e)に示すアップデートカードを作成して出力することができる。

【0053】アップデートカードは、反映テーブルのプログラム区分がHの項に対応して作成し、その項の番号

X欄の値が第2プログラム上アップデート対象を第2ソーステーブルの配列順番号で示し、フラグがCであれば第2プログラムの該当命令文をコメント文のみの行に変更する(REPLACE指令文)ことによって削除する。

【0054】又、フラグがRであれば処理対象の命令文の後に番号Y欄の値を配列順番号とする第1ソーステーブル上の命令文を挿入する(INSERT指令文)ようにアップデートカードを生成する。

【0055】即ち、反映テーブルの第2項に対応して図8(e)の第1の指令文が、フラグCに対処するように作成される。この指令文「-REPLACE 1 * A」は、配列順番号「1」の命令文を「* A」というコメント行に置き換える指令文で、コメントの中に元の命令文(A)を残して変更履歴を明らかにする。

【0056】又、反映テーブルの第6項はフラグがRであるので、第2の指令文に示すように「-INSERT 3 A」が作成され、配列順番号「3」の命令文の後に、命令文「A」(第1プログラムの配列順番号「2」の命令文)を挿入する指令文を設ける。同様に、反映テーブルの第8項、第11項に対応して、(e)の第3及び第4の指令文が得られる。

【0057】それらのアップデートカードを、図8(a)のソーステーブルに示す第2プログラムに適用すると、同じソーステーブルの「更新後」の欄に示すように更新されて、第1プログラムの更新内容を第2プログラムに反映することができる。

【0058】

【発明の効果】以上の説明から明らかなように本発明によれば、計算機の原始プログラムの更新処理において、シーケンス番号に依存しないで2プログラムの相違を移動を含めて自動認識でき、それに基づいて2つの独立した更新の一方の更新を他方に反映する制御情報を自動的に生成するので、プログラムの更新を効率よく処理することができるという著しい工業的效果がある。

【図面の簡単な説明】

【図1】 本発明の構成を示すブロック図

【図2】 テーブルの生成関係を説明する図

【図3】 最初のマッチング処理の流れ図(その1)

【図4】 最初のマッチング処理の流れ図(その2)

【図5】 2回目のマッチング処理の流れ図

【図6】 各種テーブルの例を説明する図

【図7】 反映テーブル作成処理の流れ図

【図8】 処理結果例の説明図

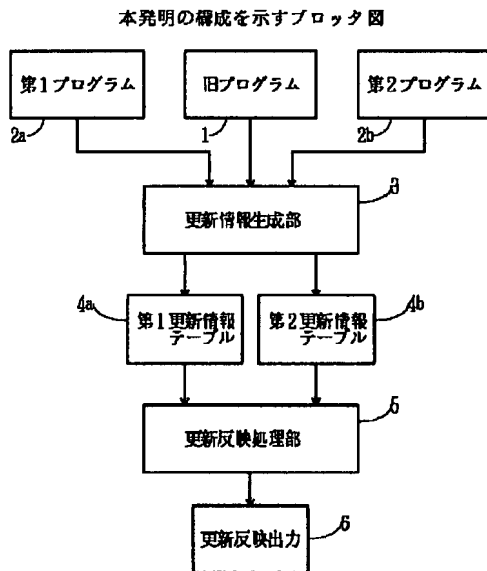
【符号の説明】

- 1 旧プログラム
- 2a 第1プログラム
- 2b 第2プログラム
- 3 更新情報生成部
- 4a 第1更新情報テーブル

9

- 4b 第2更新情報テーブル
- 5 更新反映処理部
- 6 更新反映出力
- 11 旧ソーステーブル
- 12a 第1ソーステーブル
- 12b 第2ソーステーブル
- 13 旧順位テーブル

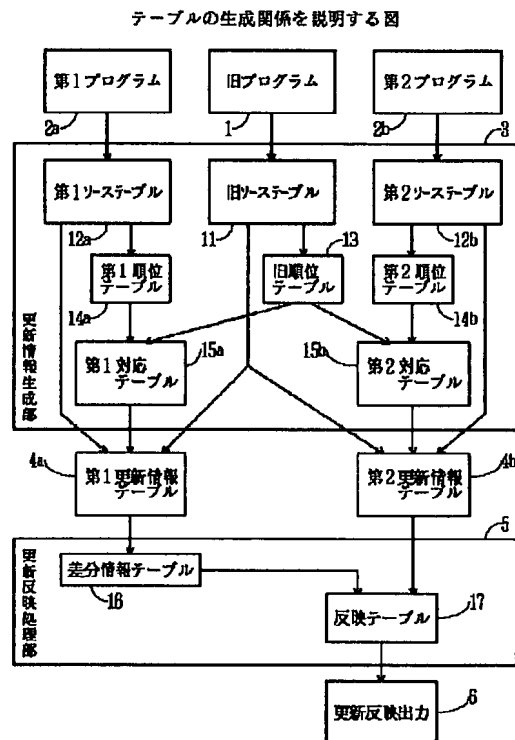
【図1】



10

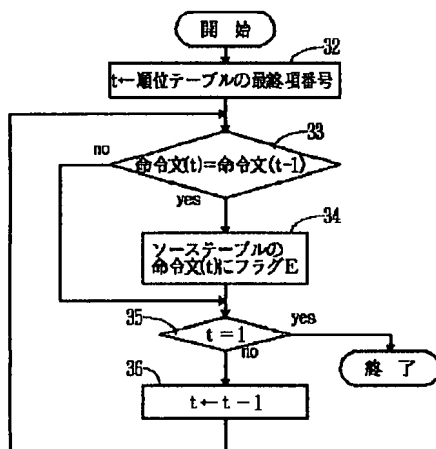
- 14a 第1順位テーブル
- 14b 第2順位テーブル
- 15a 第1対応テーブル
- 15b 第2対応テーブル
- 16 差分情報テーブル
- 17 反映テーブル
- 20~36, 40~54, 60~69 処理ステップ

【図2】



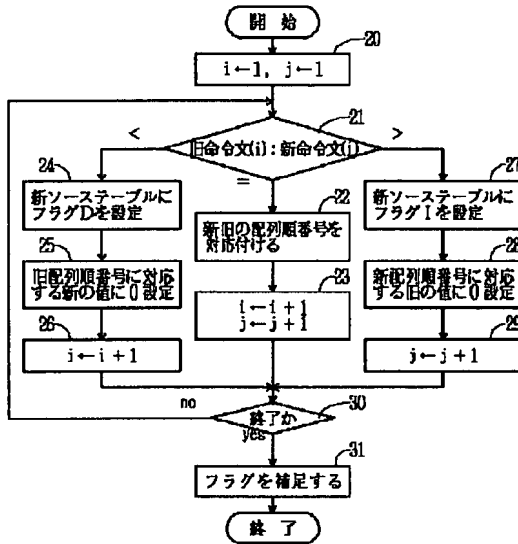
【図4】

最初のマッチング処理の流れ図 (その2)



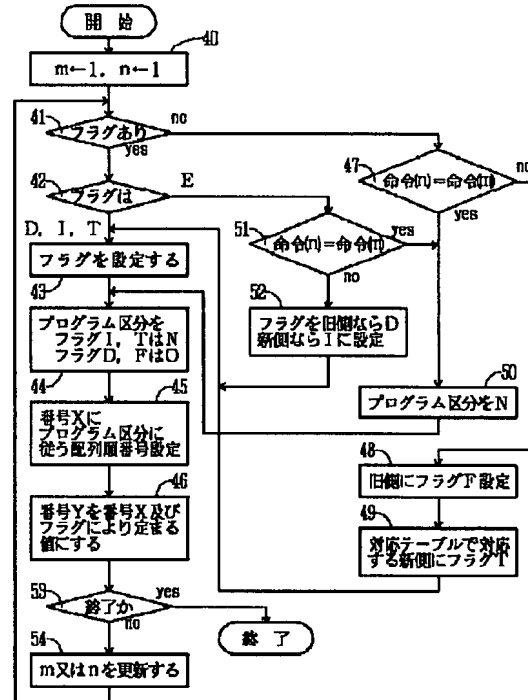
【図3】

最初のマッチング処理の流れ図 (その1)



【図5】

2 回目のマッチング処理の流れ図



【図6】

各種テーブルの例を説明する図

(a)

フラグ配列順番号	旧プログラム
1	A
2	B
3	D
4	C
5	E

フラグ配列順番号	新プログラム
1	B
2	A
3	C
4	E
5	F

(b)

旧順位	配列順番号
1	1
2	2
3	4
4	3
5	5

新順位	配列順番号
1	1
2	2
3	3
4	4
5	5

(c)

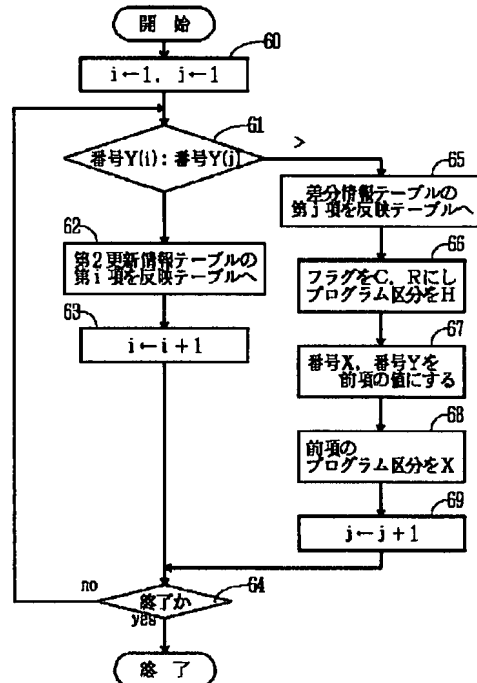
配列順番号	新に對する 旧の番号	旧に對する 新の番号
1	1	2
2	2	1
3	4	3
4	3	4
5	5	5

(d)

項	フラグ	番号X	プログラム区分	番号Y
1	F	1	O	1
2		2	N	2
3	T	3	N	3
4	D	4	O	4
5		5	N	4
6		4	N	5
7	I	5	N	5

【図7】

反映テーブル作成処理の流れ図



【図8】

処理結果例の説明図

(a) ソーステーブル

配列順番号	旧プログラム	第1プログラム	第2プログラム	更新後
1		A	B	A
2		B	A	*B
3		D	C	BN
4		C	E	D
5		E	F	*D
6			E	C
7				E
8				F

(b) 第1更新情報テーブル

項	プログラム	項フラグ番号X	区分番号Y
1	F	1	O
2		1	N
3	T	2	N
4	D	3	O
5		3	N
6		4	N
7	I	5	N

第2更新情報テーブル

項	プログラム	項フラグ番号X	区分番号Y
1		1	N
2	D	2	O
3	I	2	N
4	I	3	N
5		4	N
6		5	N
7		5	N

(c) 差分情報テーブル

項	プログラム	項フラグ番号X	区分番号Y
1	D	1	O
2	I	2	N
3	D	3	O
4	I	5	N

(d) 反映テーブル

項	プログラム	区分	項フラグ番号Y	番号X
1	X		1	1
2	O	D	2	1
3	N	I	2	2
4	X	I	2	3
5	X		3	4
6	N	C	4	5
7	X		5	6

(e) アップデートカード

-REPLACE	1
*	A
-INSERT	3
	A
-REPLACE	4
*	D
-INSERT	6
	F

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 04-242829

(43)Date of publication of application : 31.08.1992

(51)Int.Cl.

G06F 9/06

(21)Application number : 03-000457

(71)Applicant : FUJITSU LTD

(22)Date of filing : 08.01.1991

(72)Inventor : OKUZUMI NOBUAKI

MURAOKA FUMIKO

YOSHIDA SADAHIKO

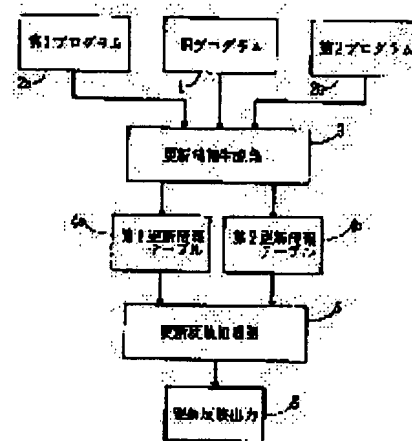
(54) PROGRAM UPDATE PROCESSING DEVICE

(57)Abstract:

PURPOSE: To reflect update of one program on the other by automatically generating control information which reflects update, to which the first program is subjected, on the second program to recognize the difference between two programs independently of the sequence number.

CONSTITUTION: When comparing an old program 1 and a first program 2a obtained by updating the old program 1 with each other to generate update information, an update information generating part 3 takes in both programs to generate an old order table and a new order table where instruction statements are arranged in order in accordance with prescribed relations among character strings of instruction statements.

Instruction statements of the old program 1 and those of the first program 2a are compared with each other in accordance with orders indicated in old and new order tables, and deletion, insertion, or non-change in the old program 1 is discriminated in accordance with a prescribed condition of relations between both instruction statements. The result is referred to generate a first update information table 4a. An update reflection processing part 5 collates them to generate the update reflection output.



LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

Copyright (C); 1998,2000 Japan Patent Office